DRM을 이용한 소프트웨어 유통 관리 시스템의 설계 및 구현

*김준옥○, *강민섭, **구윤서
*안양대학교 컴퓨터공학과
**(주)에니텍
e- mail : mskang@aycc.anyang.ac.kr

Design and Implementation of Software Distribution Management System Using DRM

*Jun-Ok Kim○, *Min-Sup Kang , **Youn-Seo Koo
*Dept. of Computer Engineering , Anyang University
**Enitec Co. Ltd.

요 약

본 논문에서는 DRM 기술을 이용하여 소프트웨어의 저작권 및 라이센스의 관리를 위한 온라인 소프트웨어 유통시스템의 설계 및 구현에 관하여 기술한다. 제안된 방법에 있어서 평문 및 인증서 정보, 그리도 전자서명 등은 AES 알고리듬을 사용하여 전송하지만, 라이센스는 사용자의 공개키 방식(RSA 알고리듬)을 이용하여 전송되기 때문에 불법 사용자에 의한 라이센스 입수가 원천적으로 봉쇄된다.

또한, 라이센스의 지속적인 관리로 인해 소프트웨어 불법 사용 및 불법 배포에 노출된 소프트웨어의 저작권을 보호하는데 매우 유효하다.

1. 서 론

21세기 정보화 사회의 급속한 발전으로 예전의 아날로그화된 컨텐츠들이 급속히 디지털화된 컨텐츠로 전환되었다. 또한 인터넷의 힘을 빌려 디지털 컨텐츠는 빠른 속도로 전파되고 이용되고 있다.

그러나 인터넷을 통한 정보의 공유가 확산됨에 따라 디지털 컨텐츠의 저작권 및 라이센스의 침해로 무분별하게 불법유포되고 있으며, 많은 디지털 컨텐츠들이 유료화 되고 있으며, 저작권 보호를 위해 DRM등과 같은 불법사용방지기술이 이용되고 있다[1].

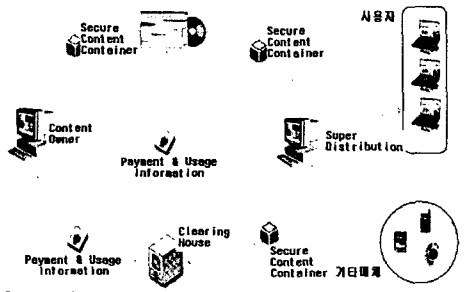
DRM은 암호화 기술을 이용하여 디지털 컨텐츠를 안전하게 보호함으로써 컨텐츠 저작권 관련 당사자의 권리 및 이익을 지속적으로 보호하고 관리하는 기술이다. 디지털 컨텐츠는 저작권자가 지정한 절차를 통해 Packaging 되어 암호화되며, 컨텐츠를 이용하고자 하는 사용자는 지정된 절차를 만족해야만 컨텐츠의 이용이 가능하다. 기존에 오프라인을 통해 유통되던 소프트웨어들이 온라인으로 유통되면서 DRM 개념이 소프트웨어도 디지털 컨텐츠의 유통 및 보호 과정에서도 적용될 수 있음을 암시하고 있다[2].

본 논문에서는 DRM 기술을 이용하여 소프트웨어의 저작권 및 라이센스의 관리를 위한 온라인 소프트웨어 유통시스템의 설계 및 구현에 관하여 기술한다. 제안된 시스템은 고속 암호알고리즘의 선택과 부분적인 패키징 기법을 고려하므로써 기존의 방법[3]과 비교할 때, 실 시간 온라인 유통 서비스가 가능하다. 또한, 제안된 방법에 있어서 평문 및 인증서 정보, 전자서명 등은 AES(Advanced Encryption Standard) [4] 알고리듬을 사용하여 전송하지만, 라이센스는 RSA 알고리듬)[5] 이용하여 전송되기 때문에 불법 사용자에 의한 라이센스 입수가 원천적으로 봉쇄된다.

2. 관련 연구

2.1 DRM(Digital Rights Management)

DRM이란 암호화 기술을, 디지털 컨텐츠가 저작자 및 유통업자의 의도에 따라 전자상거래를 통해서 안전하고 편리하게 유통업자의 의도에 따라 전자상거래를 통해서 안전하고 편리하게 유통될 수 있도록 제공되는 모든 기술과 서비스 절차 등을 포함하는 개념이다. [그림 1]은 DRM을 이용하여 디지털 컨텐츠가 유통되는 전체 과정을 나타낸다.



[그림 1) DRM을 이용한 디지털 컨텐트 유통 흐름도

그러나 디지털 컨텐츠들은 웹 상에서의 온라인으로 유통과정이 이루어 지기 때문에 보안에 매우 취약한 문제점을 가지고 있다. 이와 같은 웹의 취약점을 보완하기 위한 필수적인 기술이 바로 DRM 기술이며, 이 기술은 3가지 보안 요소를 가지고 있다.

첫 번째는 컨텐츠의 내용을 알 수 없게 엄호하는 암호화(Encryption)이고, 두 번째는 아무나 접근할 수 없게 하는 접근제한(Conditional access), 그리고 세 번째는 불법적으로 복제를 하지 못하게 하는 복제 제어(Copy Control), 복제 되었을 때 그 복제된 컨텐츠를 추적하고 확인하는 Identification과 tracing 기술이다[].

2.2 암·복호화 알고리즘

콘텐츠 암호화는 주로 대칭키 방식인 DES 또는 3DES[7] 등을 사용한다. 그러나 미 상무성 기술표준국(NIST)는 최근 계산 속도 및 안전성이 보다 강화된 차세대 암호표준으로서 RIJNDAEL 알고리듬(AES)을 채택하였다[4].

대칭키 방식은 암호화 및 복호화키가 동일하기 때문에 불특정 다수를 상대로 미리 콘텐츠를 암호화할 수 있다는 장점이 있지만, 암호화 키의 저장 및 전달 등 키 관리가 복잡하다는 단점이 있다. 한편 대칭키 방식은 콘텐츠에 대한 복호화키가 동일하기 때문에 콘텐츠의 재분배(superdistribution)를 가능하게 한다.

패키징은 암호화뿐만 아니라 기타 다른 정보, 즉 콘텐츠 타입, 제목, ID, 저작자 등과 같은 콘텐츠 및 권리자에 대한 정보를 포함한다. 또한, 콘텐츠 사용에 대한 비즈니스 룰, 사용규칙, 지불정보 등을 포함하기도 한다. 또한 각종 키 값, 라이센스를 발급받을 수 있는 위치정보(URL)를 포함한다.

암호화키는 패키징 단계에서 콘텐츠를 암호화하는데 사용하고, 복호화키는 라이센스 발급 서버(클리어링하우스)에 등록하여 허가받은 사용자에게 라이센스를 발급할 때 라이센스에 포함하여 전송한다.

본 연구에서 AES 알고리즘은 기존의 DES 알고리듬 보다 공격에 매우 강할 뿐만 아니라, 연산속도가 매우 개선된 AES를 채용하며, 라이센스의 효율적인 관리를 위하여 RSA 알고리듬을 사용한다.

2.3 SSL(Secure Socket Layer) 프로토콜

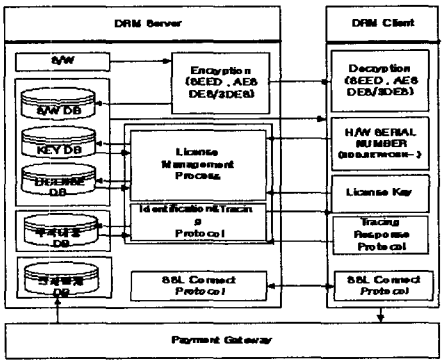
SSL은 넷스케이프에서 개발한 보안 프로토콜로 TCP/IP 위에서 송수신자사이의 통신의 비밀성과 데이터 무결성 제공을 목적으로 개발 되었다[6].

SSL은 크게 핸드쉐이크 프로토콜과 레코드 프로토콜로 되어있고, 핸드쉐이크 프로토콜은 다시 Change cipher spec protocol, Alert Protocol, handshake Protocol, application Protocol 나누어진다. SSL은 일련의 동작과정 동안 서버와 클라이언트의 진위 확인을 하도록 해 주며, 사용하는 어플리케이션에 대해 독립적이어서 HTTP나, FTP, Telnet 등의 어플리케이션이 SSL을 기반으로 운용된다. 또한 암호화 키(encryption key)와 관련된 협상을 할 수 있을 뿐만 아니라 상위의 응용 프로그램이 정보를 서버와 교환하기 전에 서버의 진위를 확인해 줄 수 있다. 암호화와 진위 확인, 메시지 확인 규칙 등의 방법을 통해 송수신 경로의 보안과 안정성을 유지시켜 준다[]. SSL은 보안 통신을 위한 메커니즘으로 가장 많이 사용하고 있으며, 안정성이 매우 높기 때문에, 본 논문에서 제안하고 있는 DRM Server와 Client와의 통신을 위해 SSL 프로토콜을 채용한다.

3. DRM 유통관리 시스템의 설계

3.1 DRM 유통관리 시스템

[그림 3]은 DRM 유통관리 시스템의 내부 구조를 나타낸다. DRM 서버는 크게 암복호 모듈, 라이센스 관리모듈, 키 관리 모듈(SSL 모듈), 거래내역 관리 모듈 등으로 구성되며, DRM Client는 라이센스 키 관리 모듈, 복호화 모듈, 그리고 추적 응답 프로토콜 등으로 구성된다.



[그림3] DRM 유통관리 시스템의 구조도

[그림 3]에서 관리 서버(DRM Sever)와 소프트웨어 내에 모듈화된 클라이언트(DRM Client)가 교환하는 정보와 통신간 프로세서와 데이터베이스를 보여주고 있다.

DRM서버는 실행 시 SSL Sever, DBMS 와 함께 실행되어 클라이언트와 상호간의 SSL 모듈을 통하여 데이터를 통신할 수 있도록 구성되어 있다. DBMS는 SSL에 의해 복호화 된 값만을 취하며, SSL은 Blocking 된 상태로 클라이언트의 Ack을 기다리도록 되어있다. 또한 서버의 어플리케이션은 DBMS의 SQL Query을 통해 라이센스의 기한 관리 및 클라이언트의 통계 및 직접적인 제어 등의 유저 인터페이스를 제공한다.

클라이언트는 제공 소프트웨어 제작 시 내부 스크립트나 내부함수로 만들어져 외부의 공격으로부터 보호한다. 또한 서버와의 통신 중에 장애발생의 경우를 고려하여, 제작 시 정책적으로 효율적인 운영 정책을 구성하여 정보를 삽입하여야 한다.

실행 시 SSL을 통하여 클라이언트의 정보를 관리서버로 제공하며, 서버의 데이터 분석 후, 제어 메시지를 통해 소프트웨어 메인 프로세스를 실행 또는 중지 할 수 있다.

3.2 SSL모듈을 통한 신뢰성 있는 제어

[그림 4]는 서버와 클라이언트의 프로세스, 데이터베이스, 인터페이스상의 데이터 흐름을 보여주고 있다. 소프트웨어는 디지털 컨텐츠보다는 라이센스의 기한도 매우 길며, 다른 시스템으로의 이동 및 시스템의 변동을 고려하여, 모든 관리와 제어를 위해 데이터베이스에 소프트웨어, 키, 라이센스, 추적로그의 데이터베이스를 각각 서버에 두고 있다.

내부 프로세스와의 흐름은 대부분 쿼리 제어문을 통하여 이루어지며, 외부의 클라이언트 통신은 SSL 프로세스를 통하여

하드웨어 정보 및 클라이언트 제어 메시지를 교환함으로써 신뢰성 있는 연결설정 및 통신으로 외부에서의 크래킹, 역어셈블과 같은 공격으로부터 소프트웨어 저작권 및 라이센스를 보호할 수 있다.

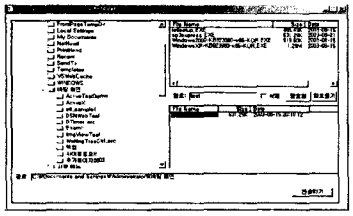
4. 시스템의 구현 결과

4.1 시스템 구현 및 환경

유통 시스템의 구현을 위하여 MS SQL SERVER 2000을 이용하여 DB를 구축하였고, 시스템과의 효율적인 인터페이스(GUI: Graph User Interface)와 DBMS 제어를 위해서 Borland 사의 C++ builder 5를 이용하였다. 또한, WEB상에서의 전자상거래에서, DBMS와의 효율적 연동을 위해 ASP(Active Server Page)를 사용하였다.

구현된 시스템은 처음 클라이언트가 삽입된 소프트웨어의 구동에서부터 시작되며, 서버는 blocking 상태로, 클라이언트의 통신을 기다리고 있다.

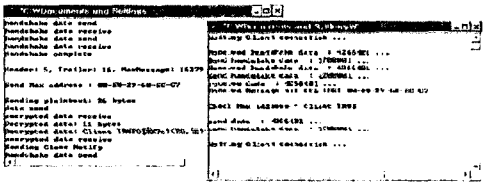
[그림 5]는 제공자로부터 전달 받은 소프트웨어 원본을 AES 암호화 알고리즘을 사용하여 패키징하는 과정을 나타낸다.



[그림 5] 전송을 위한 소프트웨어 패키징 과정

암호화 시에는 한 개 이상의 파일을 하나의 파일(.DRM)로 생성이 되며, 이 파일을 사용하기 위해서는 데이터 베이스에 저장된 서버로부터 복호화 키를 전달 받아야 한다. 사용자가 사용하는 키는 클라이언트 인식 시 데이터베이스의 중요 키 값으로 작용하므로, 클라이언트는 키에 대한 정보를 계속 저장, 유지한다. 이때, 키 값과 같은 중요 정보들은 SSL 프로토콜을 통하여 통신하게 된다.

[그림 6]은 SSL 서버와 클라이언트 상호간 프로토콜 연동결과를 나타낸다.



[그림 6] 서버와 클라이언트의 SSL 프로토콜 연동결과

SSL은 최초 핸드 쉐이크 절차를 통하여 상호간 인증을 확인한 후, 하드웨어 정보 및 제어 메시지의 전달로 서버는 클라이언트를 통제하게 된다. SSL을 통하여 교환되는 정보는 암호화 되어 보내지게 되므로 그 상호간 통신의 신뢰성을 유지할 수 있다.

[그림 7]은 유통관리 서버의 인터페이스를 나타내는 화면이며, 각각의 화면은 소프트웨어의 소스관리 창, 키 관리 창, 라이센스 관리 창, 추적 클라이언트 관리 창, 그리고 SSL 콘솔 창을 보여준다.

[그림 7]에서, 초기상태에서는 Blocking 에서 클라이언트가 동작하게 되면, SSL SERVER 모듈에서 먼저 클라이언트의 인증을 확인하고, 그 후 서버의 동작이 이루어진다. 서버는 데이터베이스의 데이터들을 이용하여, 클라이언트의 정보를 분석하여, 정상적인 라이센스인지를 검색하여, 클라이언트를 제어하며, 필요에 따라 서버에서 직접 데이터베이스를 수동으로 제어할 수 있도록 인터페이스를 구성하였다.

5. 결 론

본 논문에서는 DRM 기술을 이용하여 온라인 소프트웨어 유통 시스템의 설계 및 구현에 관하여 기술하였다. 본 논문에서 평문 및 인증서 정보등은 AES 알고리듬을, 라이센스 전송은 RSA 알고리듬을 이용하여 전송되기 때문에 불법 사용자에 의한 라이센스 입수를 원천적으로 봉쇄할 수 있다.

제안된 시스템은, 라이센스의 지속적인 관리로 인해 소프트웨어 불법 사용 및 불법 배포에 노출된 소프트웨어의 저작권을 보호하는데 매우 유효하다. 그러나 제안된 시스템은 SSL 서버에서의 순차적 처리로 인하여 다수의 사용자가 동시에 접속했을 경우 하나 이외의 클라이언트를 인식하지 못하는 문제가 발생되므로, 현재 다수의 사용자가 사용 가능하도록 현재 테스트 중에 있다.

<참고 문헌>

[1] DRM Working Group, http://www.digicaps.co.kr.

[2]AAP, Digital Rights Management for Ebooks: Publisher equipments version 1.0, 2000.

[3] DRM Working Group, http://www.fasoo.com.

[4]Advanced Encryption Standard Development Effort, http://www.nist.gov/aes.

[5] R. L. Rivest, A. Shamir and L. Adleman, " A method for obtaining digital signatures and public-key cyptosystems," Communication of the ACM, Vol. 21, No. 2, pp. 120-126, 1978.

[6] Sherif, M.H., Serhrouchni, A., Gaid, A.Y., Farazmandnia, F., " SET and SSL: Electronic Payments on the Internet," ISCC '98. Proceedings. Third IEEE Symposium, pp. 353 □ 358, 1998.

[7] NBS, " Data Encryption Standard" ,FIPS Pub, 46, U.S, National Bureau of Standard, Washington DC, Jan. 1977.